Automated reaction based on risk analysis and attackers skills in intrusion detection systems

Wael Kanoun, Nora Cuppens-Bouhlahia, Frédéric Cuppens, José Araujo

Automated Reaction based on Risk Analysis and Attackers Skills in Intrusion Detection Systems

Wael Kanoun[1,2], Nora Cuppens-Boulahia[1], Frédéric Cuppens[1] and José Araujo[2]

[1] TELECOM Bretagne, Cesson Sévigné, France
[2] Bell Labs - Alcatel Lucent, Nozay, France

Abstract

Nowadays, intrusion detection systems do not only aim to detect attacks; but they go beyond by providing reaction mechanisms to cope with detected attacks, or at least reduce their effects. Previous research works have proposed several methods to automatically select possible countermeasures capable of ending the detected attack, but without taking into account their side effects. In fact, countermeasures can be as harmful as the detected attack. Moreover, sometimes selected countermeasures are not adapted to the attacker's actions and/or knowledge. In this paper, we propose to turn the reaction selection process intelligent by giving means to (i) quantify the effectiveness and select the countermeasure that has the minimum negative side effect on the information system by adopting a risk assessment and analysis approach, and (ii) assess the skill and knowledge level of the attacker from a defensive point of view.

keywords: *Intrusion detection system, attack scenario, countermeasure, risk analysis, potentiality, impact, skill and knowledge.*

1 Introduction

In intrusion detection approach, the main objective is to detect and identify attacks or intrusions; and then react to counter them, to block them, or to mitigate their impact on the information system (IS). There are two different approaches for the reaction perspective: hot reaction [15] and policy based reaction [17, 12]. The first aims to launch a local action on the target machine to end a process, or on target network component to block a traffic, that are the cause of the launched alerts. The second acts on more general scope: it considers not only the threats reported in the alerts, but also constraints and objectives of the organization operating the IS and this by modifying the access control policy. Therefore a trade-off can be established between security objectives, operation objectives and constraints. Whatever the adopted approach, each countermeasure can have negative or positive side effects. The same countermeasure capable of ending an attack could make the IS more vulnerable, expose it to other attacks, or even have an overall impact more disastrous than the attack itself. For instance, *Firewall reconfiguration* is effective against a denial of service (DoS) attack, but can be harmful if valuable connections to a critical server could be potentially lost.

Therefore many questions emerge: Is it better to stand still? Or is the attack harmful enough to react? In this case, which countermeasure must be selected with minimum negative side effects? To answer these questions, we adopt a risk assessment and analysis approach. This approach is already used to analyze and evaluate the risks that threaten organization assets. We aim to use the same approach to evaluate the effectiveness of each countermeasure in real-time, and improve the automated reaction mechanism. The first step of a risk analysis method is to collect data that describes the system state in real-time. The second step is analyzing them and finding the potential threats and their severity. The final step is to study the countermeasure effectiveness to eliminate these threats or reduce their severity: The goal is not always to block the attack, but to minimize the risk incurred by target IS. Therefore a risk assessment method is used to evaluate and quantify the risk of an attack and its countermeasures. The method is useful to decide when it is suitable to react, and which countermeasure should be activated.

Another important aspect is the Attackers Skills and Knowledge level *(SK_Level)*. Such data is useful for the automated reaction process. If a novice script kiddie attacker is trying to establish a remote session, a simple *TCP reset* will be enough to eliminate the detected threat. Otherwise, in the case of an experienced attacker, the *TCP reset* can be ineffective and a *firewall reconfiguration* may be needed. Therefore assessing the *SK_Level* make the reaction deci-

sion module more accurate and effective. We can assume that a risk assessment and analysis approach combined with the assessment of the attackers skill and knowledge level make the automated process of reaction and countermeasure selection more accurate, realistic, cost effective, and with minimum intervention of the human administrator.

The remainder of this paper is organized as follows. To determine when a reaction is required and which is the best countermeasure to activate, our solution using risk analysis and skill and knowledge assessment approaches will be presented in section II. The implementation of our model and the conducted tests are showed in section III. In section IV, other reaction models and risk analysis models from the literature are presented. Finally section V concludes the paper.

2 Solution

To react against attacks, an efficient diagnostic procedure to detect and identify the intrusions is needed. However, due to the limitation and unreliability of the intrusion detection probes like SNORT [10], only low-level events can be detected with potentially high rates of false alarms. Therefore, to detect and recognize the current attack, an alert correlation procedure is required for proper reaction. The correlation procedure recognizes relationships between alerts in order to associate these alerts to a more global intrusion scenario, and the intrusion objectives that the attacker is seeking to accomplish. There are many approaches that can be used for this purpose: implicit [21], explicit [18, 23], and semi-explicit [14, 24] correlations. The implicit approach tries to establish implicit relations, e.g. statiscal relations, between the observed alerts or events. On other hand, in the explicit approach, whole attack scenarii are defined using explicit relations between the alerts or events. This approach is static because it requires an exhaustive definition for all the known attacks, and it is not adapted to the non-automated attacks. The semi-explicit approach is based on the description of the elementary intrusions corresponding to the alerts. The LAMBDA [16] language can be used to describe these elementary steps by defining their pre-conditions and post-conditions. This approach then finds causal relationships between these elementary alerts and connects these elementary alerts when such a relationship exists.

The correlation procedure then consists in building a scenario that corresponds to an attack graph. The attack graph is a set of paths where the nodes are the elementary attack steps; the attacker choses a path and execute these elementary steps to achieve his intrusion objective. We have decided to use this approach due to its flexibility and dynamicity: we do not have to define static exhaustive relations between the elementary steps to construct the attack scenarii, but instead, these relations can be discovered dynamically in real-time.

Semi-Explicit Correlation Definition Two LAMBDA models A and B are correlated if the post-condition of A matches the pre-condition of B. This semi-explicit [14, 24] approach is more generic and flexible than the other correlation approaches, because only the elementary steps are defined as entities and not the whole attack scenarii. A short description of the LAMBDA model fields of an elementary step is presented; for a formal description interesting readers can refer to [16]:

- Pre-conditions: This field describes the IS state required so that the attacker is able to perform the step.
- Post-conditions: This field describes the IS state after the execution of the step.
- Detection: This field is used for the mapping of a LAMBDA model to the appropriate alert.
- Verification: This field can be used to verify if a step is successfully executed.

In this paper, we will need mostly the first two fields (i.e. pre-conditions and post-conditions); examples will be given in section III.

Regarding reaction, this approach can provide a precise diagnosis of the ongoing intrusion scenario by construction the attack graph, predict the potential future steps and the intrusion objectives. Using an approach similar to the one used to describe elementary intrusions, elementary countermeasures can be specified. In this case, anti-correlation [13] can be used to find the countermeasures capable of ending a detected scenario.

Anti-Correlation Definition Two LAMBDA models A and B are anti-correlated if the post-condition of A matches the pre-condition negation of B. The anti-correlation [13] approach is based upon finding the appropriate countermeasure that turn an elementary future step of an attack unexecutable due to preconditions value modifications. Therefore, using the anti-correlation approach, the administrator knows which countermeasures from a predefined library are capable of blocking the threat.

2.1 Risk Assessment Model

As explained in the previous section, the anti-correlation approach can be used to generate a set of candidate countermeasure capable of ending the detected attack, but without assessing its impact nor the candidate countermeasure. Therefore, this reaction approach can be refined by combining it with the risk analysis model proposed in [19]. This

model is used to evaluate the total risk gravity of the IS once an attack is detected and after simulating the execution of the candidate countermeasure. Only the countermeasures that reduce the total risk gravity are kept and a new set of Risk Efficient Countermeasures *(Risk_Eff_CM)* is instantiated. The total risk gravity can be assessed after evaluating the Potentiality *(Pot)* and the Impact *(Imp)* of the detected attacks. The structure of the model is described in Figure 1. The total risk gravity of the IS is derived from the risk gravities of the detected attack scenarii. Each scenario risk gravity depends on its potentiality and impact factors. Interested readers can refer to [19] for more details.

2.1.1 Potentiality Pot

The major factor Potentiality Pot measures the probability of a given scenario to take place and achieve its objective with success. To evaluate Pot, we must first evaluate its minor factors: natural exposition $Expo$ and dissuasive measures $Diss$ and we have to take into account classification of the attack also. These minor factors can be evaluated after the appropriate audit clusters are calculated. These clusters are questions-tests that aim to evaluate the system state (active services, existent vulnerabilities, etc.). The value *zero* indicates that the studied scenario is impossible, and the value *MAX_VALUE* indicates that the occurrence and the successful execution of the scenario are inevitable.

2.1.2 Impact Imp

The second major factor to evaluate Risk Gravity of an attack scenario is Impact Imp. $\overrightarrow{Imp}$ is defined as a vector with three elements that correspond to the three fundamental security properties: Availability $Avail$, Confidentiality $Conf$ and Integrity $Integ$. Therefore, with each Intrusion Objective, a vector $\overrightarrow{Imp}$ is associated and should be evaluated. Actually, it is not possible to statically evaluate $\overrightarrow{Imp}$ of a scenario (or more precisely the $\overrightarrow{Imp}$ of the scenario's intrusion objective) directly because it depends on several dynamic elements. The impact depends on the importance of the target assets $\overrightarrow{Class}$, and the impact reduction measures level $\overrightarrow{IR}$ that are deployed on the system to reduce and limit the impact once the attack was successful.

2.1.3 Risk Gravity of an Attack Scenario or a Countermeasure $Grav$

For each detected attack, the risk gravity must be evaluated to estimate the danger level of this attack. The risk is the combination of potentiality and impact using a predefined function f. An attack that occurs frequently with little impact may have the same risk level as another rare attack that have significant impact. If a scenario has Pot or Imp equal to *zero*, the scenario's gravity risk $Grav$ will be null. To assess the risk Gravity of a candidate countermeasure CM_u, the same function f is used to assess the risk as shown in the following equation:

$$G_{CM_u} = f(Pot = \textit{MAX\_VALUE},\ Imp = CM_u.Impact) \tag{1}$$

The use of *MAX_VALUE* for the *Pot* parameter is justified by the fact that countermeasures, contrary to detected attack scenarii, do not have intrinsic potentiality. This can be explained by the fact that once a countermeasure is selected, it must be activated successfully in the IS and therefore its impact must be considered with maximum potentiality. For the attack scenarii, each one has a proper potentiality that must be combined with the attack scenario impact to deduce the risk gravity.

2.1.4 Total Risk Gravity $Total\_Grav$ and $Total\_Grav'_u$

In most situations, the correlation and reaction modules do not deal with one specific scenario. Instead, these modules have to take into account many candidate and even simultaneous scenarii. Therefore, before estimating the total gravity of risk, we must evaluate the gravity of risk of each scenario in the attack graph separately. Then we define the total gravity as an ordered vector containing the values of gravity risk of each candidate scenario. An order relation can be defined between the different instances of $\overrightarrow{Total\_Grav}$ using the lexicographic comparison. Thus, we are able to judge which attack graph has the highest risk gravity. We define also $\overrightarrow{Total\_Grav'_u}$ similarly to $\overrightarrow{Total\_Grav}$, where the difference is that $\overrightarrow{Total\_Grav'_u}$ is assessed with the new state of the IS after the simulated execution of the countermeasure CM_u, in other words using the new generated attack graph that takes into account the activation of the CM_u.

2.1.5 Risk Efficient Countermeasures Set

Once for each countermeasure u, G_{CM_u} and $\overrightarrow{Total\_Grav'_u}$ are evaluated, $\overrightarrow{Total\_Grav\_CM_u}$ can be evaluated :

$$\overrightarrow{Total\_Grav\_CM_u} = \overrightarrow{Total\_Grav'_u} \cup G_{CM_u} \tag{2}$$

where $\cup$ is a concatenation operator.

Now, only the countermeasures from $Anticorrelated\_CM$ that decrease the total risk gravity are kept and a new set $Risk\_Eff\_CM$ is defined that contains only risk efficient countermeasures:

$$\begin{gathered} \forall CM_u \in Anticorrelated\_CM; \\ \overrightarrow{Total\_Grav\_CM_u} \leq \overrightarrow{Total\_Grav} \\ \Rightarrow CM_u \in \textit{Risk\_Eff\_CM} \end{gathered} \tag{3}$$

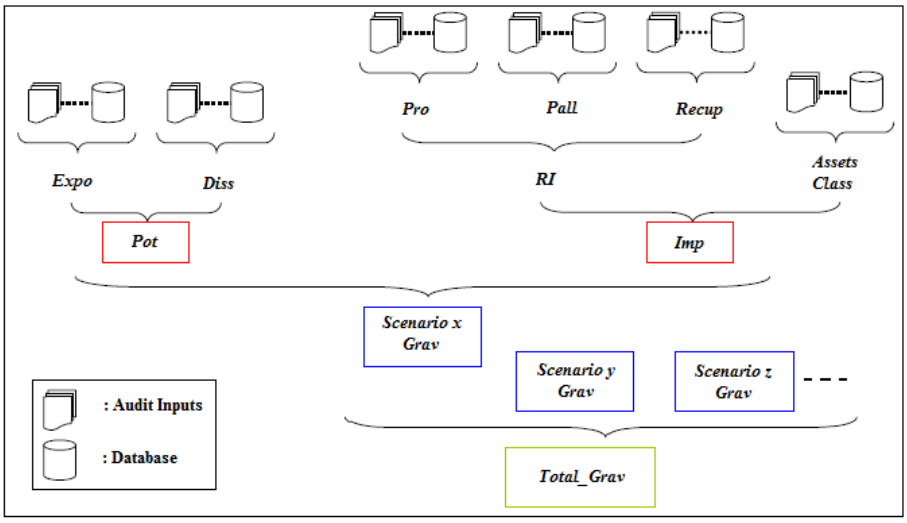

Figure 1. Risk assessment structure

2.2 Skill and Knowledge Assessment

To react properly against a detected attack scenario, the countermeasure selection process must take into account the observed skill and knowledge of the attacker. To end an attack executed by a novice, a simple *close connection* can be effective, which it is not the case when facing an expert attacker who can easy counter this reaction and a *firewall reconfiguration* is needed. In general, it could be useless to activate a complex countermeasure against a beginner; or to activate a simple countermeasure against an expert attacker who can easily bypass it. Therefore, the assessment of the attacker's skill and knowledge level would be very useful to tune our reaction model. Another point is that the attacker can have internal knowledge of the IS. For instance, a remote attacker that have the proper credentials, and thus is able to connect from the first attempt, must be taken in consideration that he/she has internal knowledge and/or high level of expertise. Another example is the case of or an attacker that is able to predict the tcp sequence of a connection that uses a complex algorithm (and not use the standard tcp sequence number incremental algorithm). As we explained before, each complex attack scenario can be modeled with an attack graph where the nodes are the elementary action steps. These elementary steps can be more or less difficult to be performed, dependently on the attacker's skills and target IS internal knowledge.

2.2.1 SK_Level Label

To assess the attacker's skill and knowledge, a defensive point of view is adopted. From the point of view of the target IS, the only information about attacker's actions lays the generated alerts that instantiates steps of the attack graph. Each step is described with a specific language like LAMBDA [16]. We propose to add a new label called Skill and Knowledge level *SK_Level*. For the attack actions, this label indicates the minimum level of skill and knowledge required to execute this action-step successfully. For the countermeasure, it indicates the value of this level that the attacker can not bypass once it is activated. This new label can have the values shown in Table 1. Using the *SK_Level* values of the executed steps retrieved from the attack graph, it is possible to assess the attacker's level of skills and determine if he/she has already internal knowledge of the target IS. Another interesting approach consists of not only considering the *SK_Level* values, but also their sequence with taking into account the time dimension; however this approach will not be considered in the present paper.

Table 1. SK_Level label values

SK_Level	**Skill**	**Internal Knowledge**
0	Low	No
1	Medium	No
2	Medium	Yes
3	High	No
4	High	Yes

2.2.2 Attacker's SK_Level and Skill and Knowledge Efficient Countermeasure Set

We consider that an attacker, capable of executing an attack step that has a high *SK_Level*, is an expert attacker;

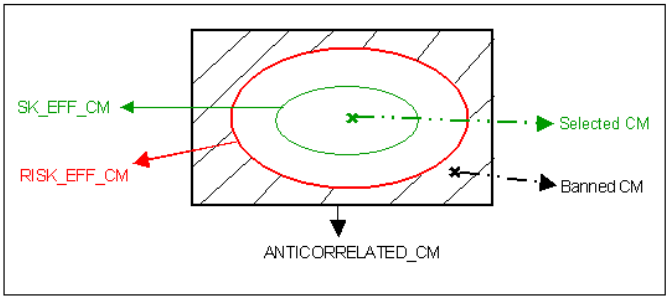

Figure 2. *Risk_Eff_CM* **and** *SK_Eff_CM* **sets**

and thus a more sophisticated countermeasure is required. The attacker's skill and knowledge Level *(SK_Level)* must be evaluated to refine the countermeasure selection. A first approach to assess the attacker's *SK_Level* is to retrieve the *SK_Level* maximum value among the successfully executed attack steps. The use of *Max* function might not be enough accurate, and further advanced approaches could be explored. Once the attacker's *SK_Level* is assessed, only the countermeasures that have a *SK_Level* greater than the attacker's one will be kept. Hence, the selected countermeasure is adapted to the attacker's Level and a new set called Skill and Knowledge Efficient countermeasures *(SK_EFF_CM)* can be instantiated.

The correlation engine by determining the executed attack steps by an attacker, is capable of assessing the attacker's level of skill and knowledge *Attacker_SK_Level*. The first approach to evaluate *Attacker_SK_Level* can be done using the following equation:

$$Attacker\_SK\_Level = Max_i(SK\_Level_i)$$

$$where\ SK\_Level_i = Attack\ Step_i.SK\_Level, \quad (4)$$

$$and\ Step_i \in Executed\_Steps$$

A possible method is to set honeypots [20, 26] and redirect the attacker to execute his/her attack steps on them. This will be used to collect the maximum number of executed steps to assess accurately his/her *SK_Level*.

Once the *Attacker_SK_Level* had been assessed, only the countermeasures from *Risk_Eff_CM* such that the *SK_Level* is higher than *Attacker_SK_Level* are kept, and a new set called *SK_Eff_CM* (see Figure 2) can be defined and instantiated countermeasures:

$$\forall CM_u \in Risk\_Eff\_CM;$$

$$Attacker\_SK\_Level \leq CM_u.SK\_Level \quad (5)$$

$$\Rightarrow CM_u \in SK\_Eff\_CM$$

2.3 Countermeasure Selection Procedure

Once the *Risk_Eff_CM* and *SK_Eff_CM* have been instantiated, a clear automatic procedure can be applied to select the most appropriate countermeasure :

```
If SK_Eff_CM ≠ ∅
   Select(Min_SK_Level(CM_u))
If Risk_Eff_CM ≠ ∅
   Select(Min_Risk(CM_u))
Select(None)
```

As suggested in this algorithm, countermeasures that belong to *SK_Eff_CM* set has the highest priority because they are adapted to the attacker's skill and knowledge level, and they are capable of reducing the overall risk gravity. If no such countermeasure could be found, the search moves to the *Risk_Eff_CM*, and the selected countermeasure is able to reduce the overall risk without being adapted to *SK_Level* of the attacker. Finally, where *Risk_Eff_CM* is empty, no countermeasure will be selected because all the candidate countermeasures are not risk aware; in other words they will increase overall risk.

3 Implementation of the Solution

CRIM (Correlation and Recognition of Malicious Intentions) [11] is a prototype that has been developed by TELECOM Bretagne. It implements the fusion, semi-explicit correlation and anti-correlation features using the LAMBDA language [16]. It collects the generated alerts and aggregates them. Then CRIM visualizes the detected attacks in real time, the future steps that can be executed by the attacker using the semi-explicit correlation principle, and the candidate countermeasure using the anti-correlation principle. As a proof of concept, a new module has been created to validate our proposal briefly described in the previous sections.

As shown in Figure 3, this module is used to assess the Risk Gravity of the detected attack scenarii and the candidate countermeasures, then it instantiates the *Risk_Eff_CM* countermeasure set. A first version of this module has been developed, but no public version is yet released. Another module takes in charge of assessing the attackers *SK_Level* and compare it to the *SK_Level* of the countermeasures that belong to the *Risk_Eff_CM*, then it instantiates the *SK_Eff_CM* set. Once the two sets are instantiated, the selection procedure can be applied to activate the most appropriate and effective countermeasure. Works are in progress to develop a first version of this module. Early tests had been performed using a LAMBDA library containing 25 step actions, 5 countermeasures, and 4 main scenarii: Code-Red attack, Trinoo attack, Sapphire attack, and Mit-

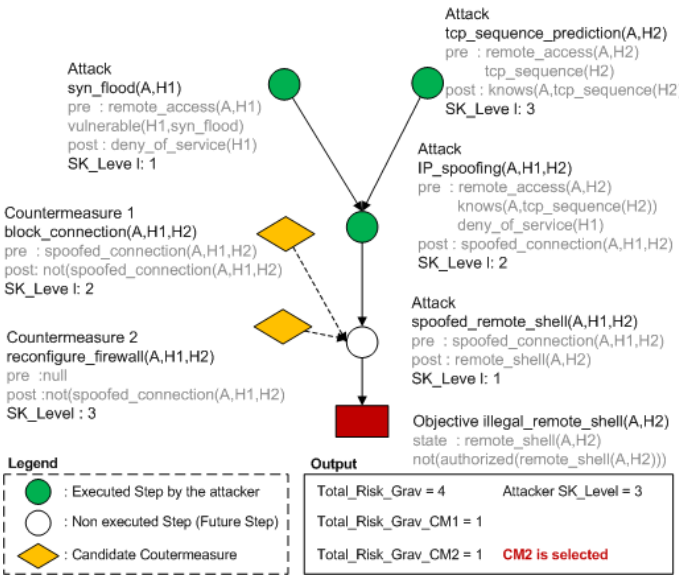

Figure 3. CRIM output: Attack graph, Intrusion objectives, Attack risk gravity, reaction risk gravity, Attacker's SK_Level

nick attack; interested readers can refer to [1] for detailed descriptions of these attack scenarii. Due to lack of space, only the last attack scenario will be presented in this paper.

The Figure 3 shows the output of the CRIM prototype corresponding to the detection of the Mitnick attack. The Mitnick attack aims to gain remote illegal shell by causing a denial of service on a legal machine and then stealing its pre-established tcp connection with the target machine. Thus, the attack graph generated by CRIM using the LAMBDA language is composed of four elementary steps: *syn_flood*, *tcp_sequence_prediction*, *ip_spoofing*, *spoofed_remote_shell*. The intrusion objective is *illegal_remote_shell* We suppose that the attacker was capable to execute successfully the first three steps. Therefore one final step remains before the attacker achieves his or her intrusion objective *Illegal Remote Shell* on a critical machine. Dark green (or gray in white and black version) circles represent elementary steps of the intrusion scenario, light yellow lozenges correspond to candidate reaction and boxes are intrusion objectives. The attacker has executed successfully the first three steps, and thus only one step remains. Therefore the potentiality and therefore the *Total_Risk_Grav* have high values (=*4*). In other hand, the attacker's *SK_Level = Max(1,3,2) = 3*.

There are two candidate countermeasures capable of reducing the total gravity risk from 4 to 1 (see *Total_Risk_CMx* in Figure 3), therefore the two countermeasures are in the *Risk_Eff_CM*. The attacker's skill and knowledge level is 3 and that indicates the fact he or she is not a novice. Thus, a *block connection* that has a *SK_Level = 2* could be not efficient and a *firewall reconfiguration* that has a *SK_Level = 3* is needed. Hence, only the second countermeasure is in *SK_Eff_CM* and recommended to be launched.

4 Related Works

Intrusion detection systems with reaction capabilities like SNORT [10] already exist. SNORT offers reflex reaction when a given attack is detected like blocking packets, sending visible warning and logging. No advanced reasoning on the reaction consequence is conducted, and side effects could appear with devastating consequences. Many industrial solutions exist like IBM Internet Security [7] and Cisco Secure IDS [3]. These solutions are efficient in intrusion prevention and offer protection against well known attacks with the corresponding impact using database like CVSS [8]. The main drawback is that there is no following up and no monitoring of the detected attacks to assess their risks and impacts in real time once they bypassed the security and prevention measures, dependably on the target organizations and assets. Another limitation is that these solutions handle vulnerabilities exploit without considering the complete scenarii. On other hand, there are several Risk Assessment methods like EBIOS [6, 2], MARION [4], MEHARI [5], etc. These methods are used to manage system assets and evaluate the risk that threatens these assets; they are unfortunately abstract, informal and incompatible with intrusion detection and computer systems: Many elements and parameters are related to physical and nature disasters (fire, earthquake, failure, etc.). There are also elements that need redefinition to be compatible with the intrusion detection systems like potentiality and impact of a threat. As suggested in [25], the risk exposure can be evaluated in terms of business perspective by using financial metrics. Another problem is that these methods are not adapted to be used in real-time. Our goal is to evaluate the system and the available countermeasure actions in real-time to help the administrator to chose the best countermeasure, and even make the reaction process automatic with minimum human intervention.

5 Conclusion and Future works

A first version of the risk analysis module has been implemented, and current works is being conducted to develop the Skill and Knowledge Assessment module. Further Series of tests will be conducted to evaluate the effectiveness and the performance of these added modules.

In the future, attack and countermeasure will be modeled with LAMBDA models, and attacks simulations will

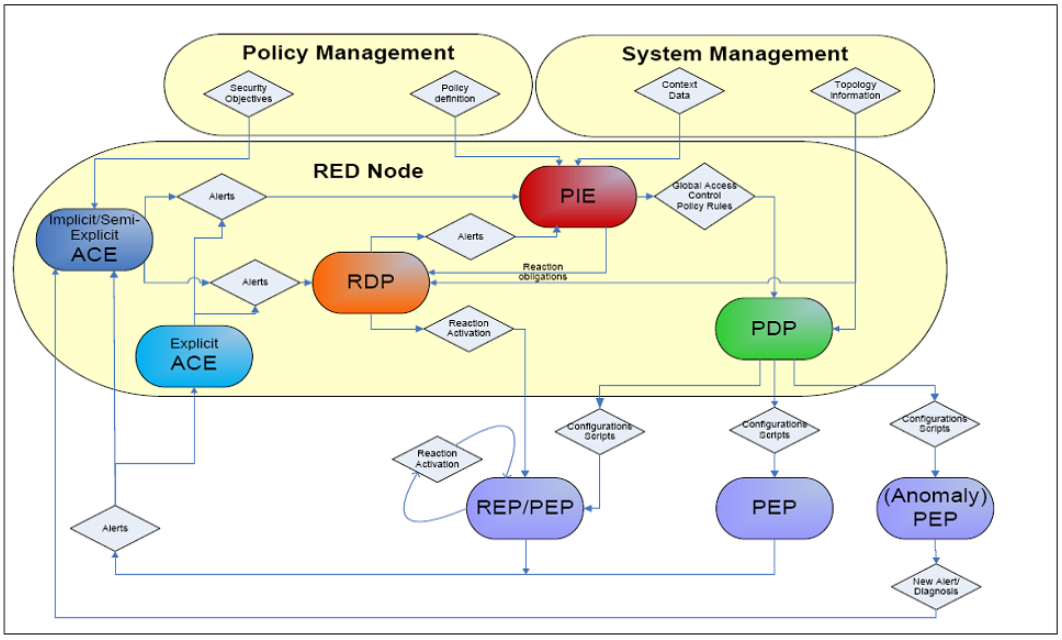

Figure 4. ReD Reaction workflow deployment

be conducted for VoIP services: CRIM will control and supervise the VoIP services status, detect and recognize attacks in real time, then conduct a Risk analysis and Skill and Knowledge assessment to propose to the administrator, or to the automated reaction module, the most effective countermeasure. This will be conducted in the ReD (REaction after detection) project [9] with other industrial partners. ReD proposes an auto adaptive model (see Figure 4) that starts from the security policy management of the monitored IS. The low level tools including intrusion detection and access control mechanisms that are implemented locally to monitor the IS are configured according to the high level security specifications. Then, according to the different alerts generated, the alerts are forwarded to the upper level whenever it is necessary, after crossing the different reaction levels, to evaluate the current system state where either direct responses are launched or the whole security policy is changed according to the detected threat. We define three levels of reaction; (1) low level reaction, (2) intermediate level reaction, and (3) high level reaction. Each level considers particular security requirements and deploys appropriate security components and mechanisms to react against the detected threats.

The risk analysis and attacker's Skill and Knowledge Assessment approach proposed in this paper will be integrated in ReD framework, and more precisely in the ACE (Alert Correlation Engine) and RDP (Reaction Decision Point) modules. The first is responsible for attack identification and diagnosis by constructing ongoing attack scenarii according to the alerts generated by the sensors. The second module decides which reactions should be triggered according to the attack scenarii received from the ACE. The selected reaction will be deployed by the REP (Reaction Enforcement Point) and PEP (Policy Enforcement Point). Our approach will turn these two modules more intelligent and risk-aware, by assessing the attacks and reaction induced risks on the entire IS. Therefore the intermediate level and high level reactions will be adapted with the organization policy and take in consideration impact and risks constraints compliantly with the organization needs and strategy. Moreover, another is being developed to communicate the attack graph, candidate countermeasures, risk analysis results, and skill and knowledge assessment results to the security console. This console will visualize graphically these data. Interested readers can refer to [9] for more details about ReD architecture.

In ReD framework, when the Mitnick attack (see section III) is detected, three reactions levels will be activated:

- Low level reaction: At this level, reflex measures like logging the attacker activity and the incoming packets, or dropping spoofed IP packets, can be activated. The activation of these reaction measures does not require high intelligence level.
- Intermediate level reaction: As presented in section III, the most efficient countermeasure will be selected to end the ongoing attack. At this level, the reaction procedure selection is more intelligent.

- High level reaction: This reaction level aims not only to end the detected attack, but also prevents the occurrence of this attack in the future by activating or redefining the security policy. In ReD project, the OrBac model [22] has been chosen as a security policy model. In OrBac, special contexts can be defined and activated when an attack is detected. In the case of Mitnick attack, the proper security policy context will be selected, and the activated security policy rules will be deployed on PEPs (e.g. firewalls, files and account rights, etc.).

Intrusion detection systems aim to detect attacks, however such detection is not quite useful without reaction. Against given attacks, there could be many possible countermeasures. Our approach will help administrators taking their decisions and selecting the proper countermeasure(s) by assessing the impact of both detected attacks and the candidate countermeasures, and taking in consideration the attacker's skill and knowledge level. As we know, no similar approach exists in the literature. We are even conducting further research and tests to turn the reaction selection and activation process fully automated.

Acknowledgment

This work was partially supported by the European CELTIC ReD project.

References

[1] Cert website: `http://www.cert.com`.

[2] Certification of EBIOS method with iso 27001: `www.cases.public.lu/publications/recherche/these_jph/NMA-JPH_MISC27.pdf`.

[3] Cisco secure systems official website: `www.cisco.com/en/US/products/sw/secursw/ps2113/index.html`.

[4] Description of MARION method published by CLUSIF: `http://i-a.ch/docs/CLUSIF_Marion.pdf`.

[5] Description of MEHARI method published by CLUSIF: `www.clusif.asso.fr/fr/production/ouvrages/type.asp?id=METHODES`.

[6] EBIOS description published by Sécréteriat Géneral de la Défense Nationale République Francaise: `www.ssi.gouv.fr/fr/confiance/documents/methodes/ebiosv2-memento-2004-02-04.pdf`.

[7] IBM internet security systems official website: `www.iss.net`.

[8] National vulnerabilities database official website: `http://nvd.nist.gov/cvss.cfm`.

[9] ReD (Reaction after Detection) project website: `http://www.celtic-initiative.org/Projects/RED/`.

[10] Snort official website: `www.snort.org`.

[11] F. Autrel and F. Cuppens. *CRIM : un module de corrélation d'alertes et de réaction aux attaques*, volume 61, chapter Annals of Telecommunications. September-October 2006.

[12] N. Cuppend-Boulahia and F. Cuppens. Specifying intrusion detection and reaction policies: An application of deontic logic. In *Ninth Workshop on Deontic Logic in Computer Science (DEON'08)*, Luxembourg, July 2008.

[13] F. Cuppens, F. Autrel, Y. Bouzida, J. Garcia, S. Gombault, and T. Sans. *Anti-correlation as a criterion to select appropriate counter-measures in an intrusion detection framework*, chapter Annals of Telecommunications. January 2006.

[14] F. Cuppens, F. Autrel, A. Miège, and S. Benferhat. Recognizing malicious intention in an intrusion detection process. In *Second International Conference on Hybrid Intelligent Systems*, Santiago, Chili, December 2002.

[15] F. Cuppens, S. Gombault, and T. Sans. Selecting appropriate counter-measures in an intrusion detection framework. In *Computer Security Foundation Workshop*, Pacific Grove, California, June 2004.

[16] F. Cuppens and R. Ortalo. Lambda: A language to model a database for detection of attacks. In *Third International Workshop on Recent Advances in Intrusion Detection (RAID'2000)*, Toulouse, France, 2000.

[17] H. Debar, Y. Thomas, F. Cuppens, and N. Cuppens-Boulahia. Enabling automated threat response through the use of a dynamic security policy. *Journal in Computer Virology (JCV)*, 3, August 2007.

[18] M. Huang. A large-scale distributed intrusion detection framework based on attack strategy analysis. Louvain-La-Neuve, Belgium, 1998.

[19] W. Kanoun, N. Cuppens-Boulahia, and F. Cuppens. Advanced reaction using risk assessment in intrusion detection systems. In *Second International Workshop on Critical Information Infrastructures Security*, Malaga, Spain, 2007.

[20] S. Krasser, J. Grizzard, and H. Owen. The use of honeynets to increase computer network security and user awareness. School of Electrical and Computer Engineering.

[21] R. Lippmann. Using key string and neural networks to reduce false alarms and detect new attacks with sniffer-based intrusion detection systems. In *Second International Workshop on the Recent Advances in Intrusion Detection (RAID'99)*, October 1999.

[22] A. Miège. *Définition d'un environnement formel d'expression de politiques de sécuérité: Modèle OrBAC et extensions*. PhD thesis.

[23] B. Morin and H. Debar. Correlation of intrusion symptoms: an application of chronicles. In *Proceedings of the Sixth International Symposium on the Recent Advances in Intrusion Detection (RAID'02)*, Pittsburg, USA, September 2003.

[24] P. Ning, Y. Cui, and D. S. Reeves. Constructing attack scenarios through correlation of intrusion alerts. In *ACM Conference on Computer and Communications Security*, 2002.

[25] J.-P. Sauvé, R. A. Santos, R. R. Almeida, and J. A. B. Moura. On the risk exposure and priority determination of changes in it service management. *Lecture Notes in Computer Science*, pages 147–158, September 2007.

[26] C. Yin, M. Li, J. Ma, and J. Sun. Honeypot and scan detection in intrusion detection system. School of Electronic InformationEngineering.